AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior revisions, and listings, of claims in the application.

Listing of Claims:

1.   (Amended) A computer readable medium containing instructions for enabling a subject matter expert to develop a custom software solution for a desired automation project including at least one solution process, comprising:

a subject matter expert interface operative to enable the subject matter expert to enter and develop rules for modeling the solution process;

fluid object material operable to develop a morphable object and source code based on the rules input through the subject matter expert interface; and

a user interface based on the morphable object, and being operative to enable a user to enter known information into the morphable object for solving problems within the domain of the modeled solution process, said user interface changing in appearance in response to said user entered information.

2.   (Original) The computer readable medium according to claim 1 further comprising a knowledge database operable to store the rules input through the subject matter expert interface, and wherein the subject matter expert interface is further operable, through the use of empty fields, to guide the subject matter expert through the entry of variables, templates, references, and events to develop the rules.

3.   (Original) The computer readable medium according to claim 1 further comprising an instantiation modeler operable to instantiate and morph the morphable object as known information is entered by the user.

4.   (Original) The computer readable medium according to claim 3 wherein the instantiation modeler is operable to repeatedly instantiate the morphable object as known information is entered by the user, the instantiation modeler is operable to morph the morphable object into progressively more mature morphable objects based on previously instantiated objects, and the user interface is operable to change based on the more mature instantiated object.

5.   (Original) The computer readable medium according to claim 1 wherein the fluid object material is provided with a wrapper operable to permit a subject matter expert to pause operation and update rules.

6.   (Original) The computer readable medium according to claim 5 wherein the subject matter expert interface is further operative to enable the subject matter expert to enter, develop, and change to, new and/or revised rules simultaneous with the user interface guiding the user through entry of known information, and the wrapper is operable to briefly pause operation while the morphable object is revised.

7.   (Original) The computer readable medium according to claim 6 wherein the subject matter expert interface and fluid object material reside on a development server, and the user interface resides on a user terminal.

8.   (Original) The computer readable medium according to claim 6 wherein the fluid object material is further operable to develop a revised morphable object based on the revised rules input through the subject matter expert interface, the user interface is further operable to revise based on the revised morphable object.

9.   (Original) The computer readable medium according to claim 1 wherein the fluid object material is operable to develop a plurality of morphable objects forming coded cells and coded contexts, and further comprising an instantiation modeler operable to instantiate the morphable objects and morph the morphable objects based on known information input by the user and based on the coded cells and coded contexts.

10.   (Original) The computer readable medium according to claim 1 wherein the fluid object material comprises plurality of abstraction levels.

11.   (Original) The computer readable medium according to claim 1 wherein the fluid object material comprises at least five abstraction levels.

12.   (Original) The computer readable medium according to claim 1 further comprising a code generator operable to generate source code implementing the fluid object material and the source code being based on the rules.

13.   (Original) The computer readable medium according to claim 12 wherein the generated source code is complete.

14.   (Original) The computer readable medium according to claim 12 wherein the code generator is further operable to generate source code implementing the fluid object material, the fluid object material including a plurality of morphable objects, and the code generator is still further operable to generate source code in separate segments.

15.   (Original) The computer readable medium according to claim 12 further comprising a wrapper operable to transmit a compiled version of the source code from a development server to a production server.

16.   (Original) The computer readable medium according to claim 1 wherein the fluid object material comprises a plurality of metadata levels.

17.   (Original) The computer readable medium according to claim 1 wherein the morphable object is substantially completely comprised of metadata and processed metadata.

18.   (Original) The computer readable medium according to claim 1 wherein the subject matter expert interface is further operable to guide the subject matter expert through the entry of variables and templates, and the subject matter expert interface provides a plurality of inheritance levels among templates.

19.   (Original) The computer readable medium according to claim 1 wherein the subject matter expert interface is further operable to guide the subject matter expert through the entry of variables and templates, and the subject matter expert interface provides at least five of inheritance levels among templates.

20.   (Amended) A method for implementing automation in a computer system, the method comprising:

> enabling a subject matter expert to input rules;
>
> inputting rules;
>
> forming a knowledge database based on the rules;
>
> ~~developing~~ generating source code for a fluid object and creating a morphable object based on the rules; and

enabling a user to input known information into the morphable object through a user interface ,wherein appearance of said user interface varies in response to information input by said user .

21.   (Original) The method according to claim 20 further comprising instantiating the morphable object when a piece of known information is input into the morphable object, and morphing the morphable object based on the piece of known information.

22.   (Original) The method according to claim 20 further comprising repeatedly instantiating the morphable object as known information is input into the morphable object, and repeatedly morphing the morphable object based on the known information.

23.   (Original) The method according to claim 22 wherein repeatedly instantiating the morphable object as known information is input into the morphable object and repeatedly morphing the object based on the known information comprise automatically repeatedly instantiating the morphable object as known information is input into the morphable object and automatically repeatedly morphing the object based on the known information.

24.   (Original) The method according to claim 20 wherein forming a knowledge database based on the rules, developing the morphable object based on the rules, and enabling the user to input known information into the morphable object comprise automatically forming the knowledge database based on the rules, automatically developing the morphable object based on the rules, and automatically enabling the user to input known information into the morphable object.

25.   (Original) The method according to claim 20 wherein forming a knowledge database based on the rules and developing the morphable object based on the rules comprise automatically forming the knowledge database based on the rules and automatically developing the morphable object based on the rules.

26.   (Original) The method according to claim 20 wherein forming a knowledge database with the rules comprises automatically forming the knowledge database based on the rules.

27.   (Original) The method according to claim 20 wherein developing the morphable object based on the rules comprises automatically developing the morphable object based on the rules.

28.   (Original) The method according to claim 20 wherein enabling the user to input known information into the morphable object comprises automatically developing a dynamic user interface for input of known information.

29.   (Original) The method according to claim 20 further comprising generating source code for implementing the fluid object material and the morphable object, and compiling the source code.

30.   (Original) The method according to claim 29 wherein generating the source code comprises automatically generating the source code upon subject matter expert indication to generate.

31.   (Original) The method according to claim 29 further comprising transmitting the compiled source code from a development server to a production server.

32.   (Original) The method according to claim 31 further comprising implementing an instantiation modeler with the compiled source code, instantiating the morphable object when a piece of known information is input into the morphable object, and morphing the morphable object into a mature morphable object based on the piece of known information.

33.   (Original) The method according to claim 32 further comprising morphing the mature morphable object when another piece of known information is input into the mature morphable object.

34.   (Original) The method according to claim 32 further comprising repeatedly morphing the mature morphable object when additional pieces of known information are input into the mature morphable object.

35.   (Original) The method according to claim 20 further comprising updating the rules in the knowledge database while the user inputs known information into the morphable object.

36.   (Original) The method according to claim 35 further comprising generating and compiling revised source code for the updated rules, pausing operation of the user interface, updating a previous code with the revised code, and releasing the user interface for continued operation.

37.   (Original) The method according to claim 20 wherein enabling a subject matter expert to input rules comprises enabling the subject matter expert to define a plurality of metadata points about a piece of data and define a plurality of metadata processes about the piece of data.

38.   (Original) The method according to claim 37 wherein enabling a subject matter expert to input rules further comprises enabling the subject matter expert to define second level metadata points about the metadata points and to define third level metadata points about the second level metadata points.

39.   (Original) The method according to claim 37 wherein one of the metadata points is an empty field for a piece of data.

40.   (Original) A computer system comprising:

a development server including fluid object material operable to develop a morphable object based on rules input through a subject matter expert interface;

a subject matter expert terminal in operative communication with the development server, and the subject matter expert terminal including the subject matter expert interface which is operable to guide a subject matter expert through entry and development of the rules for modeling a solution process; and

a user terminal in operative communication with the development server, and the user terminal including a user interface based on the morphable object, and being operative to guide a user through entry of known information into the morphable object for solving problems within the domain of the modeled solution process.

41.   (Original) The computer system according to claim 40 further comprising a production server in operative communication with the development server and the user terminal, such that the production server is interposed between the development server and the user terminal, and the production server including an instantiation modeler operable to instantiate the morphable object as known information is entered by the user.

42. (Original) The computer system according to claim 41 further comprising a plurality of user terminals in operative communication with the production server, a plurality of subject matter expert terminals in operative communication with the development server, and a wherein the operative communication between the development server and the production server is through a local area network.

43. (Original) The computer system according to claim 40 wherein the development server further includes a knowledge database operable to store the rules input through the subject matter expert interface.

44. (Original) A method for enabling implementation of automation, the method comprising:

enabling a subject matter expert to input rules;

enabling automatic formation of a knowledge database based on the rules;

enabling automatic development of a morphable object based on the rules; and

enabling a user to input known information into the morphable object.

45. (Original) The method according to claim 44 further comprising enabling instantiation of the morphable object when a piece of known information is input into the morphable object, and enabling morphing of the morphable object based on the piece of known information.

46. (Original) The method according to claim 45 wherein enabling morphing comprises enabling morphing by pulling in a list and enabling morphing by retrieving a look up value.

47. (Original) The method according to claim 45 wherein enabling morphing comprises enabling morphing by evaluating a prerequisite and by adding an information request based upon the prerequisite being met.

48. (Original) The method according to claim 44 wherein enabling the subject matter expert to input rules comprises enabling the subject matter expert to input data routings.

49. (Original) The method according to claim 48 wherein enabling the subject matter expert to input data routings comprises enabling the subject matter expert to input data routings based on metadata.

50. (Original) A modeler implemented by a computer program comprising instructions for:

> directing a subject matter expert through the entry of rules;
>
> formation of a knowledge database based on the rules;
>
> development of a morphable object based on the rules; and
>
> enabling a user to input known information into the morphable object.

51. (Canceled)

52. (Canceled)

53. (Canceled)

54. (Canceled)

55. (Original) A method for generating code, the method comprising:

> developing a plurality of templates containing rules;

inheriting a second one of the plurality of templates from a first one of the plurality of templates; and

generating code based on the templates.

56. (Original) The method according to claim 55 further comprising inheriting a third one of the plurality of templates from the second one of the plurality of templates, and inheriting a fourth one of the plurality of templates from the third one of the plurality of templates.

57. (Original) The method according to claim 55 further comprising inheriting a third one of the plurality of templates from the first one of the plurality of templates.

58. (Original) A method for generating code, the method comprising:

directing a subject matter expert through the entry of rules;

forming a knowledge database based on the rules;

developing a plurality of morphable objects based on the rules; and

generating code based on the morphable objects.

59. (Original) The method according to claim 58 wherein directing the subject matter expert through the entry of rules comprises directing the subject matter expert with empty fields, and developing the plurality of morphable objects comprises developing the plurality of morphable objects and forming coded cells and coded contexts.

60. (Canceled)

61. (Canceled)

62. (Canceled)

63.   (Canceled)

64.   (Original) A computer readable medium containing instructions for enabling a subject matter expert to code blank cells and blank contexts, comprising:

providing at least one blank cell and at least one blank context;

receiving knowledge from a subject matter expert;

filling the at least one blank cell with the knowledge thereby forming a coded cell;

and

filling the at least one blank context with the knowledge thereby forming a coded context.

65.   (Original) The computer readable medium according to claim 64 further comprising selecting at least one coded cell determined to contain metadata pertaining to user input, known information.

66.   (Original) The computer readable medium according to claim 64 further comprising instantiating the at least one selected coded cell, and morphing the at least one coded cell.

67.   (Original) The computer readable medium according to claim 64 further comprising selecting at least one coded context determined to contain metadata pertaining to user input known information.

68.   (Original) The computer readable medium according to claim 64 wherein the coded cells are operative to process independent variables, and the coded contexts are operative to process dependent variables.

69.   (Canceled)

70.   (Canceled)

71.   (Canceled)

72.   (Canceled)

73.   (Canceled)

74.   (Original) A method for enabling a subject matter expert through entry of knowledge into entry fields to define the parameters of a data structure residing on a computer readable medium, the method comprising:

    providing an entry field for dimension;

    providing an entry field for data type dimension;

    providing a hierarchical entry field; and

    providing a range bound entry field.

75.   (Canceled)

76.   (Canceled)

77.   (Canceled)

78.   (Canceled)

79.   (Canceled)

80.   (Canceled)

81.   (Canceled)

82.   (Canceled)

83.   (Canceled)

84.   (Canceled)

85.   (Canceled)

86.   (Canceled)

87.   (Canceled)

88.   (Canceled)

89.   (Canceled)

90.   (Canceled)

91.   (Canceled)

92.   (Canceled)

93.   (Original) A method of writing code without the assistance of a software engineer, the method comprising:

enabling a subject matter expert to input rules;

inputting rules;

forming a knowledge database based on the rules; and

automatically generating source code for implementing fluid object material.

94.   (Canceled)

95.   (Canceled)

96.   (Canceled)

97.   (Canceled)

98.   (Canceled)

99.   (Canceled)

100.   (Canceled)

101.   (Canceled)

102.   (Canceled)

103.   (Canceled)

104.   (Canceled)

105.   (Canceled)

106.   (Canceled)

107.   (Canceled)

108.   (Canceled)

109.   (Canceled)

110.   (Canceled)

111. (Canceled)

112. (Canceled)

113. (Canceled)

114. (New) A method in a computing environment for automated generation of program solutions for at least one process, the method comprising:

providing an interface for a subject matter expert to develop a plurality of rules for modeling the at least one process;

providing a code generator to generate source code to implement a fluid object material, wherein said fluid object material is developed utilizing one or more of said plurality of rules;

providing an instantiation modeler to utilize said fluid object and instantiate one or more morphable objects, wherein said morphable objects are utilized in a user interface;

wherein said morphable objects respond to information provided by a user and change the display of said user interface during execution of the program solutions.

115 (New) The method as recited in claim 114 wherein said plurality of rules may be revised by a subject matter expert, wherein revising said plurality of rules morphs said one or more morphable objects, and wherein the morphed one or more morphable objects change the appearance of said user interface.

116 (New) <u>The method as recited in claim 114 wherein the fluid object material comprises a plurality of abstraction levels.</u>

117.(Newl) <u>The method as recited in claim 114 wherein said code generator is operable to generate source code in separate segments.</u>